

Input Aggregated Network for Face Video Representation

Zhen Dong¹, Su Jia², Chi Zhang¹, Mingtao Pei¹

1. Beijing Laboratory of IIT, School of Computer Science, Beijing Institute of Technology, Beijing, China
2. Department of Applied Mathematics and Statistics, Stony Brook University, Stony Brook, USA

Abstract

Recently, deep neural network has shown promising performance in face image recognition. The inputs of most networks are face images, and there is hardly any work reported in literature on network with face videos as input. To sufficiently discover the useful information contained in face videos, we present a novel network architecture called input aggregated network which is able to learn fixed-length representations for variable-length face videos. To accomplish this goal, an aggregation unit is designed to model a face video with various frames as a point on a Riemannian manifold, and the mapping unit aims at mapping the point into high-dimensional space where face videos belonging to the same subject are close-by and others are distant. These two units together with the frame representation unit build an end-to-end learning system which can learn representations of face videos for the specific tasks. Experiments on two public face video datasets demonstrate the effectiveness of the proposed network.

1 Introduction

Video-based face recognition under uncontrolled environments is a challenging task due to large intra-class variations in pose, lighting, and facial expressions. There are lots of video-based face recognition methods to deal with these problems [Wang *et al.*, 2015; Huang *et al.*, 2015a; Hamm and Lee, 2008; Shakhnarovich *et al.*, 2002]. These methods often contain three steps: extracting frame features, modeling face video clip, and learning classifier. The three steps are independently conducted and might not be optimally compatible, *i.e.*, the frame feature and the model of a face video might not be discriminative enough for the final recognition task. Take, for example, the covariance descriptor [Huang *et al.*, 2015a; Li *et al.*, 2015b], a usual method of face video representation. The covariance features involve two procedures: extracting low-level features for each frame separately, and fusing these features by computing a covariance matrix, as shown in Figure 1(a). Both of the procedures are designed independently without taking the final task into account, which will degrade the performance.

As a quite effective tool for building an end-to-end learning system, deep learning has shown promising performances in face recognition. The representatives are Deep Face [Taigman *et al.*, 2014; Taigman *et al.*, 2015], DeepID series [Sun *et al.*, 2013; Sun *et al.*, 2014; Sun *et al.*, 2015b], and FaceNet [Schroff *et al.*, 2015]. However, all these works use face images as input, and there is hardly any work reported in literature dedicated to design network with face videos as input. To take full advantage of the useful information contained in videos, we propose the input aggregated network to learn face video representations.

The input aggregated network inputs are variable-length face videos, and outputs are fixed-length representations of the videos. As shown in Figure 1(b), the proposed network contains three units: frame representation unit, aggregation unit, and mapping unit. In the frame representation unit, a deep CNN is used to learn representation of each frame. Previous studies [Krizhevsky *et al.*, 2012; Oquab *et al.*, 2014] show that the CNN features perform better than many hand-crafted features, so the frame representation unit is able to provide good features for the aggregation unit. The aggregation unit aims at modeling several frame features of a face video as one Riemannian manifold point. To achieve this goal, four layers are involved in this unit: minus mean layer, transpose fully connected layer, outer product layer, and group average pooling layer. With these four layers, we will show that the symmetric positive definite manifold [Arsigny *et al.*, 2007] and Grasmann manifold [Hamm and Lee, 2008] can be characterized by the aggregation unit. The mapping unit serves as the map from the low-dimensional Riemannian manifold to high-dimensional space where similar face videos are close by and dissimilar faces are far away. Considering the map is usually reversible, a deep auto-encoder is expected to well approximate the map owing to its powerful ability on describing highly complex nonlinear function. These three units are optimally combined in the input aggregated network to build an end-to-end learning system where each unit is compatible with others for the final task.

To evaluate the effectiveness of the proposed network, we do experiments of face identification on two datasets: IJB-A [Klare *et al.*, 2015] and VIPL-TV [Li *et al.*, 2015a]. On both dataset, the open-set problem of face identification is considered, *i.e.*, the probe image may not in the gallery set. In the

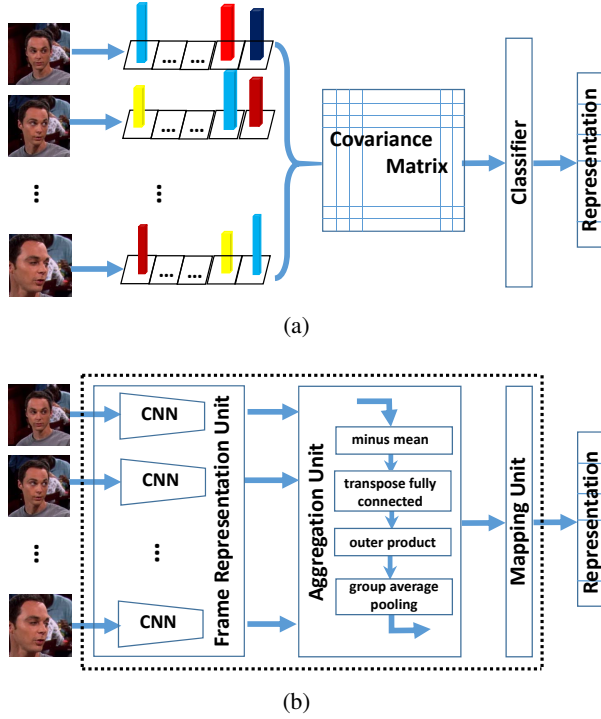


Figure 1: The illustrations of the traditional face video recognition method and our method. The traditional method (a) has three uncorrelated steps: extracting frame features, modeling face video clip, and learning classifier. Only the classifier learning procedure treats the final recognition task as optimal principle. Differently, the input aggregated network (b) integrates frame representation unit, aggregation unit, and mapping unit into an end-to-end system to learn the mapping from face videos to representations, and all the units serve for the final task.

IJB-A dataset, this is accomplished by removing some videos of specific subjects from the gallery set, and the VIPL-TV dataset has an “unknown” class which contains all face videos belonging to none of any other classes. Results on these two datasets show better performance of the input aggregated network than the state-of-the-art methods. Furthermore, the proposed network is able to learn face video representations for other various video-based face related tasks, such as face verification, face video retrieval, by designing loss functions for the specific tasks.

The contributions of the paper are two-folds: (1) The input aggregated network is able to generate fixed-length representations for variable-length face videos. (2) Based on the input aggregated network, a face video identification method is proposed and achieves the comparable performances with other state-of-the-art methods.

The remainder of this paper is organized as follows: Sec.2 reviews the related work, including face video representation methods and deep learning on face recognition. In Sec.3, we elaborate the architecture and learning policy of the proposed input aggregated network. Sec.4 shows that the symmetric

positive definite and Grassmann manifolds can be involved in the framework of the proposed network. The experiments and discussions are presented in Sec.5, and Sec.6 concludes this paper.

2 Related Works

2.1 Face Video Representation

Face video representation has been widely studied in recent years, and lots of works are proposed. Most of them treat the face video as a set of frame images, *i.e.* image set based classification. Closely-related to representation, another issue is the metric definition between image sets. One prevalent method represents face image set by a Gaussian distribution [Arandjelovic *et al.*, 2005; Shakhnarovich *et al.*, 2002], and the Kullback-Leibler divergence is used to measure the similarity between image sets. The estimated model by this kind of method will not properly fit the real word data distribution when real world data doesn’t follow the Gaussian assumption. Other methods use subspace learning technology to describe face image sets, including linear subspace and manifold. The principle angle between subspaces is often used as the metric in linear subspace based methods, such as Mutual Subspace Method [Yamaguchi *et al.*, 1998], Orthogonal Subspace Method [Oja, 1983], and their variants [Kim *et al.*, 2007; Li *et al.*, 2009]. The linear subspace methods may not work well when the face sets contain complex data variations. Instead, some methods assume that the face images lie on a nonlinear manifold and propose several manifold distance metrics, such as manifold-manifold distance [Wang *et al.*, 2012b], manifold distance analysis [Wang and Chen, 2009], covariance discriminative learning [Wang *et al.*, 2012a]. These methods treat frame feature extraction and set modeling as independent procedures, which is different from the input aggregated network. The input aggregated network builds an end-to-end learning system which involves these two procedures to learn good representations for face videos.

2.2 Deep Learning on Face Recognition

Recently, deep learning has shown promising performances on face recognition. Taigman *et al.* [2014; 2015] proposed the “DeepFace” which addresses the face verification problem by a nine-layer deep neural network with 120 million parameters, and reduced the error of the state-of-the-art by more than 25% on the LFW dataset. Sun *et al.* [2015b; 2015a] proposed the “DeepID” series which achieve exciting results on both LFW and YouTube Faces datasets for both face identification and verification. Schroff *et al.* [2015] trained a deep CNN called “FaceNet” with nearly 200M face images of about 8M subjects for face recognition and clustering. The FaceNet maps face image to a compact representation in Euclidean space and achieves a new record accuracy on both LFW and YouTube Faces datasets. Chen *et al.* [2015] aims at building an end-to-end system involving face detection, face association, landmark detection, representation components for face verification with deep convolutional neural network. Although these works have conducted experiments on face video datasets, such as YouTube Faces, and

IJB-A datasets, the networks in these works still take face frame image as input and the relationship between frames is ignored. Lu *et al.* [2015] modeled face image set as manifold, and proposed a multi-manifold deep learning method to learn the maps from face set space to a shared feature subspace. The distance between different set can be learned, but the representation of face video cannot be learned. Different from above networks, our network aims at learning a fixed-length representation of variable-length face video for various face-related tasks, such as identification, verification, and retrieval.

3 Input Aggregated Network

To represent a face video, three steps are required: representing each face frame, modeling the video clip, and mapping the video representation for the specific task. Corresponding to the three steps, the input aggregated network contains three units: frame representation unit, aggregation unit, and mapping unit.

3.1 Frame Representation Unit

As described in Sec.2.2, deep CNN has shown impressive performance on face image representation in recent years, and previous studies [Krizhevsky *et al.*, 2012; Oquab *et al.*, 2014] show that the CNN features perform better than many hand-crafted features. Motivated by these work, a deep CNN is used as for learning features of each frame in the face video. As a representative of deep CNN in face recognition, the DeepID [Sun *et al.*, 2014] significantly promoted performances than previous methods. The DeepID contains convolutional layers, fully connected layers, ReLU activation layers, and max-pooling layers. The ReLU activation layers are behind convolutional layers or fully connected layers. For simplicity, we use \mathbf{L}_{1-4} to represent the 4 convolutional layers, and $\mathbf{L}_{5,6}$ describe the 2 linear layers. The outputs of \mathbf{L}_5 are features with the dimension of 160. The \mathbf{L}_6 is followed by a softmax classifier to generate probability distribution for classification, and the dimension of features in \mathbf{L}_6 is the same with the number of class.

In the frame representation unit, we use the similar architecture as DeepID and make three main modifications. First, we change the input size from 39×31 to 48×48 for more information from larger image. Second, serving as the feature extractor, a fully connected layer $\mathbf{L}_{5.5}$ with 32 dimension is added between \mathbf{L}_5 and \mathbf{L}_6 to reduce the computation burden of the next two stages. Third, the convolutional layers \mathbf{L}_1 , \mathbf{L}_2 and fully connected layers \mathbf{L}_5 , $\mathbf{L}_{5.5}$ before ReLU layers are initialized as [He *et al.*, 2015] to accelerate the convergence. The deep CNN is pre-trained on the large-scale face image dataset CASIA-WebFace [Yi *et al.*, 2014] which has 494,414 face images of 10,575 subjects to obtain good initializations for face representation. Please note that the used DeepID-like network has a relatively simple architecture for fast computation, and other deeper network with higher performance such as VGG-Net [Simonyan and Zisserman, 2014] and GoogLeNet [Szegedy *et al.*, 2014] can also be used in this unit instead of the DeepID-like architecture.

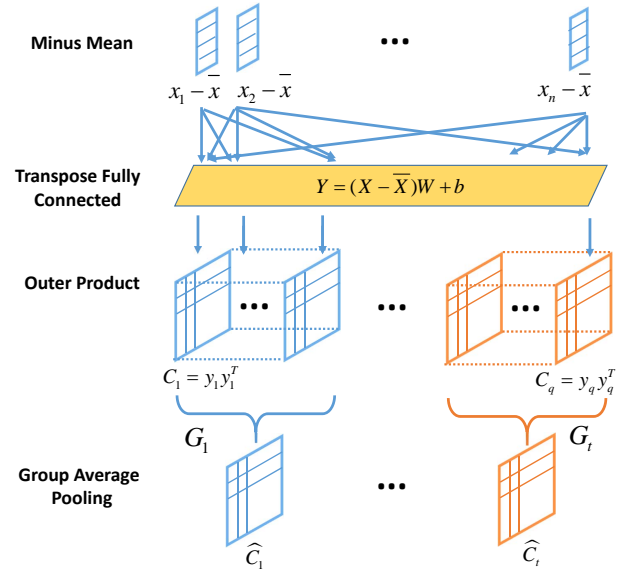


Figure 2: The architecture of the aggregation unit. The aggregation unit contains four layers: minus mean layer, transpose fully connected layer, outer product layer, and group average pooling layer.

3.2 Aggregation Unit

The aggregation unit aims at modeling the variable-length frame representations as fixed-length Riemannian manifold points. The architecture of the aggregation unit is shown in Figure 2. The aggregation unit contains four layers: minus mean layer, fully connected layer, outer product layer, and group average pooling layer.

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ be the CNN features of the n frames in a face video from the frame representation unit. The mean vector of these features are calculated as

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i. \quad (1)$$

The **Minus Mean Layer** makes the features from frame representation unit to be zero-mean by subtracting the mean vector from all the features as $[\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \dots, \mathbf{x}_n - \bar{\mathbf{x}}]$. The followed **Transpose Fully Connected Layer** is similar to fully connected layer which provides a linear transformation to its inputs, the difference is that the transpose fully connected layer conducts the linear transformation on the transpose of its input matrix, which aims at finding the optimal linear combination of its inputs to form the subspace of the input mean-subtracted features:

$$\mathbf{Y} = (\mathbf{X} - \bar{\mathbf{X}})\mathbf{W} + \mathbf{b}, \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{n \times q}$ and $\mathbf{b} \in \mathbb{R}^{d \times q}$ are the parameters of the transpose fully connected layer, and $\bar{\mathbf{X}} \in \mathbb{R}^{d \times n}$ is mean matrix whose columns are all $\bar{\mathbf{x}}$. The transpose fully connected layer is able to characterize the respective importance of the frame features in the video by the corresponding coefficients, and can be implemented by a convolutional layer with kernel

size of 1×1 . For each transformed feature \mathbf{y}_i , the **Outer Product Layer** computes the outer product of \mathbf{y}_i and itself:

$$\mathbf{C}_i = \mathbf{y}_i \otimes \mathbf{y}_i = \mathbf{y}_i \mathbf{y}_i^\top, \quad i = 1, 2, \dots, n, \quad (3)$$

where $\mathbf{C}_i \in \mathbb{R}^{d \times d}$ is a symmetric positive semi-definite matrix. The **Group Average Pooling Layer** converts the variable-lengths feature into fixed-length features by partition $\mathbf{C}_i (i = 1, 2, \dots, n)$ into t groups according to its frame order, and operating average pooling in each group along with the frame dimension.

$$\hat{\mathbf{C}}_i = \frac{1}{|\mathcal{G}_i|} \sum_{\mathbf{C}_j \in \mathcal{G}_i} \mathbf{C}_j, \quad i = 1, 2, \dots, t, \quad (4)$$

where \mathcal{G}_i is the i -th group of the matrices. Since there are n matrices coming from the outer product layer, each group in the group pooling layer process $|\mathcal{G}_i| = n/t$ matrices. The group are consecutive in the frame dimensionality, so the aggregation unit is able to hold temporal information for face representation in a certain degree. With these four layers, the aggregation unit maps n CNN frame representations as input to t matrices as features, where n is the length of variable-length video and t is the number of the fixed-length features. The t matrices are used as the input of the followed mapping unit. Since there are four parameters \mathbf{W} , \mathbf{b} , q , and t , an aggregation unit can be denoted as $\mathcal{A} = \{\mathbf{W}, \mathbf{b}, q, t\}$.

The overall input aggregated network is expected to be an end-to-end system. Since the aggregation unit forms a directed acyclic graph, it can be trained by back propagation algorithm which propagates the gradients of the loss function to the four layers. For i -th group of the group average pooling layer, let the gradient respect to $\hat{\mathbf{C}}_i$ be $\partial l / \partial \hat{\mathbf{C}}_i$, the gradient respect to \mathbf{y}_j is

$$\frac{\partial l}{\partial \mathbf{y}_j} = \frac{1}{|\mathcal{G}_i|} \left[\left(\frac{\partial l}{\partial \hat{\mathbf{C}}_i} \right)^\top + \frac{\partial l}{\partial \hat{\mathbf{C}}_i} \right] \mathbf{y}_j, \quad (5)$$

$$\mathbf{C}_j \in \mathcal{G}_i, \quad i = 1, 2, \dots, t.$$

Eq.(5) gives the gradients of outer product and group average pooling layers. Since the transpose fully connected layer is implemented by the convolutional operation with kernel size of 1×1 , the gradient of this layer is easy to obtain similar to the convolutional layer. The minus mean layer holds gradient unchanged for the reason that only a constant vector is subtracted in this layer. With the computed gradients, the aggregation unit is able to be embedded into the input aggregated network for end-to-end training.

The matrices from the outer product layer are symmetric positive semi-definite, and the average pooling in each group holds this property. The outputs of the group average pooling are even symmetric positive definite matrices when $n/t \gg d$. As proved in [Arsigny *et al.*, 2007], the space of symmetric positive definite matrices form a Lie group which is a Riemannian manifold. Therefore, the output t matrices are all lie on a Riemannian manifold. In Sec.4, we will show that several usually used manifolds in face video representation can be involved in the framework of the aggregation unit.

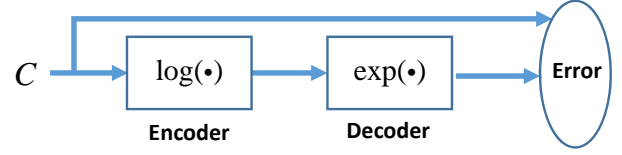


Figure 3: The illustration of auto-encoder used for pre-training the mapping unit.

3.3 Mapping Unit

Obtaining the Riemannian manifold points which represent face videos, previous methods usually map these data points into a reproducing kernel Hilbert space by kernel methods. The mapping unit in the input aggregated network aims to learn the map by the deep network with multiple layers. Let \mathcal{R} be the Riemannian manifold, and \mathcal{S} be the high-dimensional space, the goal of the this unit is learning the map

$$\mathbf{F} : \mathcal{R} \rightarrow \mathcal{S}. \quad (6)$$

The mapping function is highly complex and nonlinear in general, so we use a deep neural network with h stacked hidden layers to model the mapping. Let f_i be the mapping function of the i -th layer, and \mathbf{z}_i be the output feature representation of the i -th layer, we thus have

$$\mathbf{z}_i = \begin{cases} f_{i-1}(\mathbf{z}_{i-1}) = \tau(\mathbf{W}_{i-1}\mathbf{z}_{i-1} + \mathbf{b}_{i-1}), & i = 2, 3, \dots, h+1, \\ \mathbf{C}, & i = 1, \end{cases} \quad (7)$$

where $\mathbf{C} \in \mathcal{R}$ is the input point on the Riemannian manifold, \mathbf{W}_{i-1} and \mathbf{b}_{i-1} are the parameters of the $(i-1)$ -th layer, and τ represents the non-linear activation function, such as sigmoid, tanh, and ReLU.

Considering that the map for \mathcal{R} is usually reversible, a deep auto-encoder is trained to obtain a good initializations for the map. For example, the maps from the symmetrical positive definite Riemannian manifold to its tangent space is $\log(\cdot)$, and the inverse map is $\exp(\cdot)$ as shown in Figure 3. The cost function of the auto-encoder is

$$\min_{\mathbf{F}, \mathbf{F}'} \frac{1}{N} \sum_{j=1}^N \|\mathbf{C}^{(j)} - \mathbf{F}'(\mathbf{F}(\mathbf{r}^{(j)}))\|_2^2 + \eta \sum_k \|\mathbf{W}_k\|_F^2, \quad (8)$$

where $\mathbf{F} = \{f_1, f_2, \dots, f_h\}$ is the encoder, \mathbf{F}' is the decoder, $\mathbf{C}^{(j)}$ is the j -th point in the training set, and N is the size of training set.

4 Relation to Manifolds

In this section, we will show that several usually used methods of face video representation can be involved in the framework of the input aggregated network.

Symmetric Positive Definite Manifold: All the symmetric positive definite matrices form a Riemannian manifold [Arsigny *et al.*, 2007]. In computer vision community, the covariance matrix descriptor which is a special case of symmetric positive definite matrix is often used for face video

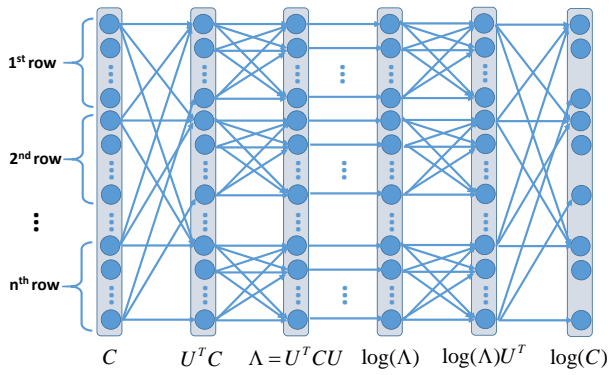


Figure 4: The mapping unit is able to represent the matrix logarithm operator $\log(\cdot)$. Please note that not all links are drawn for the sake of simplicity.

representation owing to its second order statistic property. Given a set of features \mathcal{S} , the covariance matrix descriptor is calculated as

$$\mathbf{C} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} (s - \mu_{\mathcal{S}})(s - \mu_{\mathcal{S}})^{\top}, \quad (9)$$

where $\mu_{\mathcal{S}} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} s$ represents the mean vector of \mathcal{S} . The diagonal elements of the matrix characterize the variance of each feature dimension, and the off-diagonal elements encode the respective correlations between feature dimensions. The distance between two covariance matrices, \mathbf{C}_a and \mathbf{C}_b is measured by the Log-Euclidean distance which is a geodesic distance calculated with Euclidean computations in the domain of matrix logarithms:

$$d(\mathbf{C}_a, \mathbf{C}_b) = \|\log(\mathbf{C}_a) - \log(\mathbf{C}_b)\|_F, \quad (10)$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, and $\log(\cdot)$ as the matrix logarithm operator is computed as

$$\log(\mathbf{C}) = \mathbf{U} \log(\Lambda) \mathbf{U}^{\top}, \quad (11)$$

where $\mathbf{C} = \mathbf{U} \Lambda \mathbf{U}^{\top}$ with $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$ is the eigen decomposition of \mathbf{C} , and $\log(\Lambda) = \text{diag}(\log(\lambda_1), \log(\lambda_2), \dots, \log(\lambda_d))$.

The covariance matrix descriptor can be written in the form of aggregation unit by $\mathcal{A} = \{\mathbf{I}, \mathbf{O}, d, 1\}$ where $\mathbf{I} \in \mathbb{R}^{d \times d}$ is the identity matrix, and $\mathbf{O} \in \mathbb{R}^{d \times n}$ is a matrix with all elements as 0. As shown in Figure 4, the non-linear mapping, matrix logarithm operator $\log(\cdot)$, is able to be represented by the mapping unit since \mathbf{U} and \mathbf{U}^{\top} performs linear transformation which is able to be implemented by the fully connected layer, and the feasibility of approximating the logarithm operation of number by neural network is ensured by the universal approximation theorem [Hornik *et al.*, 1989].

Grassmann Manifold: Given the feature set $\mathbf{X} \in \mathbb{R}^{d \times n}$, the Grassmann manifold represents the feature set by a q -dimensional linear subspace which is spanned by an orthonormal basis matrix $\mathbf{Y} \in \mathbb{R}^{d \times q}$:

$$\mathbf{X} \mathbf{X}^{\top} \approx \mathbf{Y} \Lambda \mathbf{Y}^{\top}, \quad (12)$$

where Λ and \mathbf{Y} are eigenvalue matrix and eigenvector matrix corresponding to the q largest eigenvalues. The distance between two points on the Grassmann manifold, \mathbf{Y}_a and \mathbf{Y}_b is denoted as

$$d(\mathbf{Y}_a, \mathbf{Y}_b) = \frac{1}{\sqrt{2}} \|\mathbf{Y}_a \mathbf{Y}_a^{\top} - \mathbf{Y}_b \mathbf{Y}_b^{\top}\|_F. \quad (13)$$

The Grassmann manifold can also be represented by the framework of the input aggregated network. Let $\mathbf{W}^* \in \mathbb{R}^{n \times q}$ be the transformation matrix from \mathbf{X} to \mathbf{Y} :

$$\mathbf{W}^* = \arg \min_{\mathbf{W}} \frac{1}{2} \|\mathbf{X} \mathbf{W} - \mathbf{Y}\|_F^2, \quad (14)$$

which can solve as $\mathbf{W}^* = \mathbf{X}^{\dagger} \mathbf{Y}$ where \mathbf{X}^{\dagger} is the pseudo-inverse of \mathbf{X} , and we have that $\mathbf{X} \mathbf{W}^* \approx \mathbf{Y}$. The \mathbf{b}^* is set as $\bar{\mathbf{X}} \mathbf{X}^{\dagger} \mathbf{Y}$, and the group size is set as 1. The output of the aggregation unit is

$$\begin{aligned} & \frac{1}{n} [(\mathbf{X} - \bar{\mathbf{X}}) \mathbf{W}^* + \mathbf{b}^*] [(\mathbf{X} - \bar{\mathbf{X}}) \mathbf{W}^* + \mathbf{b}^*]^{\top} \\ &= \frac{1}{n} \mathbf{X} \mathbf{W}^* \mathbf{W}^{*\top} \mathbf{X}^{\top} = \frac{1}{n} \mathbf{Y} \mathbf{Y}^{\top}, \end{aligned} \quad (15)$$

which is the interchangeably representation of Grassmann manifold with ignoring the constant factor $1/n$. Therefore, the Grassmann manifold is represented by the aggregation unit as $\mathcal{A} = \{\mathbf{X}^{\dagger} \mathbf{Y}, \bar{\mathbf{X}} \mathbf{X}^{\dagger} \mathbf{Y}, q, 1\}$.

5 Experiments

5.1 Datasets and Experimental Settings

We do experiments on two face video datasets: IARPA Janus Benchmark-A (IJB-A) [Klare *et al.*, 2015], and VIPL-TV [Li *et al.*, 2015b].

The IJB-A dataset¹ contains 500 subjects, and each subject has at least 5 images and 1 video to provide a total of 5,392 images and 2,085 videos, with an average of 11.4 images and 4.2 videos per subject. The IJB-A dataset doesn't provide the original videos, and the sampled frames from the videos are provided. The sampled frames are 20,412 in total. The example frames of the IJB-A dataset is shown in Figure 5. The image and videos are all under uncontrolled environments and with challenging factors, such as variations in face orientation, pose, pose, and lighting conditions.

The VIPL-TV dataset contains two large scale video collections from two hit American shows, *i.e.*, the Big Bang Theory (BBT) and Prison Break (PB). These two TV series are quite different in their filming styles. The BBT is a sitcom with 5 main characters, and most scenes are taken indoors during each episode of about 20 minutes long. Differently, many shots of the PB are taken outside during the episodes with the length of about 42 minutes, which results in a large range of different illumination. All the face video shots are

¹This paper makes use of the following data made available by the Intelligence Advanced Research Projects Activity (IARPA): IARPA Janus Benchmark A (IJB-A) data detailed at <http://www.nist.gov/itl/iad/ig/facechallenges.cfm>. The original videos are contained in the extend version Janus Challenging set 2 (JANUS CS2) which is not publicly available yet.



Figure 5: The example frames of the IJB-A dataset.



Figure 6: Example face tracks in VIPL-TV datasets, (a) is from the big bang theory, and (b) is from prison break.

collected from the whole first season of both TV series, *i.e.*, 17 episodes of BBT, and 22 episodes of PB, and the number of shots of the two sets are 4,667 and 9,435, respectively. Figure 6 shows example face tracks of this dataset.

On both datasets, our network is trained in three stages: pre-training the frame representation unit, pre-training the mapping units, and fine-tuning the whole network. In all three stages, the base learning rate is set as 0.01 and reduced by polynomial strategy with power equals to 0.05. The momentum and weight decay for optimization are set as 0.9 and 0.0005, respectively. All the codes are implemented by using Caffe deep learning toolbox [Jia *et al.*, 2014], and the experiments are conducted on a Titan-X GPU with 12GB memory. In the first stage, we pre-train the frame representation unit on the WebFace dataset with the ratio of training sample number to validation sample number as 9 : 1. The batch size is set as 128, and the total iteration number is set as 5 million. In the second stage, the mapping unit is pre-trained with the \mathbf{W} and \mathbf{f} of the aggregation unit as identity and zero matrices, respectively. The auto-encoder described in Sec.3.3 is trained where the batch size is set as 12 (videos), and the total iteration number is 200K. In the third stage, the whole network is fine-tuned with batch size and iteration number of 12 and 50K, respectively. We use two newly published manifold learning methods for face video recognition as our baseline, they are Projection Metric Learning on Grassmann manifold (PML) [Huang *et al.*, 2015b] and Discriminative Analysis on Riemannian manifold of Gaussian distributions (DARG) [Wang *et al.*, 2015]. For fair comparisons, the features from the frame representation unit are used to test these two methods.

Table 1: Results on IJB-A Dataset.

Methods	Accuracy (%)
PML	30.64
DARG	34.71
Our Method	46.25

Table 2: Results on VIPL-TV Dataset.

Methods	Accuracy (BBT) (%)	Accuracy (PB) (%)
PML	46.20	16.47
DARG	48.59	17.11
Our Method	60.32	25.81

5.2 IJB-A Dataset

The IJB-A evaluation protocol consists of identification over 10 splits. In each split, 333 subjects are randomly chosen to be training set, and the remaining 167 subjects are for testing. Every testing subject has images or frames randomly sampled into either probe set or gallery set. The image in gallery set is used as query, and the gallery set represents the imagery contained in an operational database. To meet the real word applications, the open-set identification protocol is provided, *i.e.*, the subject in probe set may not appear in the gallery set. The IJB-A dataset achieves this by randomly removing 55 subjects from the gallery set.

Considering that the IJB-A dataset only contains the sampled frames of face videos and images, and the sampled frames are not consecutive, we treat a face video as a face image set, *i.e.*, the temporal relationship among frames are not taken into account. Accordingly, the group size in the aggregation unit is set as 1. To augment the training data, we form a batch as 16 frames or images of a subject. The comparison results are shown in Table.1, the reported results are the average of ten splits. The proposed method outperforms other two baseline methods, and the main reason is that the end-to-end training strategy ensures all the three units of the input aggregated network serving for the final identification task.

5.3 VIPL-TV Dataset

Similar to the IJB-A dataset, the open-set identification protocol is used in the experiments of VIPL-TV dataset.. In both BBT and PB of the the dataset, there is an 'unknown' class which consists of subjects that don't belong to any other classes. We train the network with the 'unknown' subjects as a single class to accomplish the open-set identification. The experimental results shown in Table.2 demonstrate the effectiveness of the proposed network.

6 Conclusions

A novel network architecture called input aggregated network is proposed for face video representation. The proposed network contains three units: frame representation unit, aggregation unit, and mapping unit. These three units form an end-to-end learning system which can learn fixed-length representations for variable-length face videos for specific tasks. Experiments conducted on two public face video datasets demonstrate the effectiveness of the proposed network in face identification. The future work is to apply the input aggregated

network to other video-based face-related tasks, such as face verification and face video retrieval.

References

- [Arandjelovic *et al.*, 2005] Ognjen Arandjelovic, Gregory Shakhnarovich, John Fisher, Roberto Cipolla, and Trevor Darrell. Face recognition with image sets using manifold density divergence. In *CVPR*, volume 1, pages 581–588. IEEE, 2005.
- [Arsigny *et al.*, 2007] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 29(1):328–347, 2007.
- [Chen *et al.*, 2015] Jun-Cheng Chen, Rajeev Ranjan, Amit Kumar, Ching-Hui Chen, Vishal Patel, and Rama Chellappa. An end-to-end system for unconstrained face verification with deep convolutional neural networks. In *ICCV Workshops*, pages 118–126. IEEE, 2015.
- [Hamm and Lee, 2008] Jihun Hamm and Daniel D Lee. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *ICML*, pages 376–383. ACM, 2008.
- [He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
- [Hornik *et al.*, 1989] Kurt Hornik, Maxwell Stinchcombe, and Herbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.
- [Huang *et al.*, 2015a] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Face recognition on large-scale video in the wild with hybrid euclidean-and-riemannian metric learning. *Pattern Recognition*, 48(10):3113–3124, 2015.
- [Huang *et al.*, 2015b] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *CVPR*, 2015.
- [Jia *et al.*, 2014] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *International Conference on Multimedia*, pages 675–678. ACM, 2014.
- [Kim *et al.*, 2007] Tae-Kyun Kim, Josef Kittler, and Roberto Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *TPAMI*, 29(6):1005–1018, 2007.
- [Klare *et al.*, 2015] Brendan F Klare, Emma Taborsky, Austin Blanton, Jordan Cheney, Kristen Allen, Patrick Grother, Alan Mah, Mark Burge, and Anil K Jain. Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In *CVPR*. IEEE, 2015.
- [Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.
- [Li *et al.*, 2009] Xi Li, Kazuhiro Fukui, and Nanning Zheng. Boosting constrained mutual subspace method for robust image-set based object recognition. In *IJCAI*, pages 1132–1137, 2009.
- [Li *et al.*, 2015a] Yan Li, Ruiping Wang, Zhiwu Huang, Shiguang Shan, and Xilin Chen. Face video retrieval with image query via hashing across euclidean space and riemannian manifold. In *CVPR*. IEEE, 2015.
- [Li *et al.*, 2015b] Yan Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. Hierarchical hybrid statistic based video binary code and its application to face retrieval in tv-series. In *FG*. IEEE, 2015.
- [Lu *et al.*, 2015] Jiwen Lu, Gang Wang, Weihong Deng, Pierre Moulin, and Jie Zhou. Multi-manifold deep metric learning for image set classification. In *CVPR*, pages 1137–1145. IEEE, 2015.
- [Oja, 1983] Erkki Oja. *Subspace methods of pattern recognition*, volume 6. Research Studies Press, 1983.
- [Oquab *et al.*, 2014] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, pages 1717–1724. IEEE, 2014.
- [Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, June 2015.
- [Shakhnarovich *et al.*, 2002] Gregory Shakhnarovich, John W Fisher, and Trevor Darrell. Face recognition from long-term observations. In *ECCV*, pages 851–865. Springer, 2002.
- [Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [Sun *et al.*, 2013] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Hybrid deep learning for face verification. In *ICCV*, pages 1489–1496. IEEE, 2013.
- [Sun *et al.*, 2014] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *NIPS*, pages 1988–1996, 2014.
- [Sun *et al.*, 2015a] Yi Sun, Ding Liang, Xiaogang Wang, and Xiaoou Tang. Deepid3: Face recognition with very deep neural networks. *arXiv preprint arXiv:1502.00873*, 2015.
- [Sun *et al.*, 2015b] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deeply learned face representations are sparse, selective, and robust. In *CVPR*, June 2015.
- [Szegedy *et al.*, 2014] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.
- [Taigman *et al.*, 2014] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708. IEEE, 2014.
- [Taigman *et al.*, 2015] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Web-scale training for face identification. In *CVPR*, 2015.
- [Wang and Chen, 2009] Ruiping Wang and Xilin Chen. Manifold discriminant analysis. In *CVPR*, pages 429–436. IEEE, 2009.
- [Wang *et al.*, 2012a] Ruiping Wang, Huimin Guo, Larry S Davis, and Qionghai Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *CVPR*, pages 2496–2503. IEEE, 2012.
- [Wang *et al.*, 2012b] Ruiping Wang, Shiguang Shan, Xilin Chen, Qionghai Dai, and Wen Gao. Manifold–manifold distance and its application to face recognition with image sets. *TIP*, 21(10):4466–4479, 2012.
- [Wang *et al.*, 2015] Wen Wang, Ruiping Wang, Zhiwu Huang, Shiguang Shan, and Xilin Chen. Discriminant analysis on riemannian manifold of gaussian distributions for face recognition with image sets. In *CVPR*, pages 2048–2057. IEEE, 2015.

- [Yamaguchi *et al.*, 1998] Osamu Yamaguchi, Kazuhiro Fukui, and Ken-ichi Maeda. Face recognition using temporal image sequence. In *FG*, pages 318–323. IEEE, 1998.
- [Yi *et al.*, 2014] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.